1

Distributed Function Computation in Wireless Sensor Networks*

Lei Ying, R. Srikant, and Geir E. Dullerud

Coordinated Science Lab
University of Illinois at Urbana-Champaign
Urbana, IL 61801

Summary. A wireless sensor network is often built for a specific purpose, and sensors are required to collaborate to accomplish a common task. Many of these tasks can be regarded as computations of functions of the sensor measurements. In this chapter, we study such a function computation problem in wireless sensor networks under two different scenarios.

In Section 1.2, we consider a wireless sensor network consisting of n sensors. Each sensor has a recorded bit, which has been set to either "0" or "1," and the statistics of sensor measurements are assumed to be unavailable. The network has a special node called the fusion center whose goal is to compute a symmetric function of these measurements. We also assume the wireless channels are binary symmetric channels with a probability of error p. Under this setting, we study distributed function computation algorithms which use multi-reception diversity to combat channel noise and data aggregation to reduce the number of transmissions. We first show a trivial lower bound on the transmission energy consumption, and then propose a distributed algorithm whose energy consumption is only a factor of $\log\log n$ more than the lower bound.

In Section 1.3, we consider a different scenario where n sensors are densely deployed, and the sensor measurements are highly correlated. It is assumed that the statistics of sensor measurements are known, and the sensors can hear each other. Further, we assume that each sensor measurement can take one of m values, and each sensor also has m signals to represent those measurements. Different signal is assumed to associate with different transmission cost. The goal of the the network is to transmit information to the fusion center so that the fusion center can compute a function based on the sensor measurements. Assume that real-time function computation is required, so at each time slot, the measurement should be reported immediately. We propose a stochastic control approach to exploit the correlation of sensor measurements to achieve the minimum cost real-time function computation.

1.1 Introduction

With the wide availability of inexpensive wireless technology and sensing hardware, wireless sensor networks are expected to become commonplace because of their broad range of potential applications. A wireless sensor network consists of sensors

* The research was supported by a Vodafone Fellowship

that have sensing, computation and wireless communication capabilities. Each sensor monitors the environment surrounding it, collects and processes data, and when appropriate transmits information so as to cooperatively achieve a global detection objective. One important feature of wireless sensor networks is that the network is often designed for a specific purpose, and the sensors are required to collaborate to achieve a global objective. This is one fundamental distinction between wireless networks used for communication and wireless networks used for sensing. In wireless communication networks, the protocols are designed so that they are not application-specific, and therefore the network can support a constantly evolving set of applications. Contrasting this, in sensor networks, the architecture and protocols can be designed for each specific application, exploiting its structure, to reduce the energy usage within the network. Note that the objectives of sensor networks are to retrieve useful information from sensor measurements, so many of these objectives can be regarded as a function computation of sensor measurements. For example, counting the number of intruders or hot-spots are equivalent to computing a "sum" function, and detecting abnormal event could be same as computing a threshold function. Recently, there have been a lot of interests in function computation in wireless sensor networks. For example, in [5], the authors have designed a block coding scheme to compress the amount of information to be transmitted in a sensor network computing some functions. In [8], [14] and [15], energy consumption of function computation is studied for large scale sensor networks, where the energy consumption of each bit is assumed to be same. Also, there are several recent works [2], [1] and [16] discussed using silence to convey information in sensor networks to save energy, and indicate the potential benefit of non-traditional communication in sensor networks.

In this chapter, we consider two different scenarios. In Section 1.2, we consider a multi-hop networks with noisy communication channels where the measurement of each sensor consists of one bit; the goal is for the fusion center to compute symmetric functions — those functions determined by the sum of the observed bits. To achieve this, we would like to design a distributed algorithm while minimizing the total transmission energy consumed by the network. Specifically, distributed symmetric function computation with binary data, which is also called a counting problem in this chapter, is as follows: each node is in either state "1" or "0", and the fusion center needs to decide, using information transmitted from the network, the number of sensors in state "1". Since the wireless channels are unreliable, we adopt the multi-user diversity idea introduced in [4], where the network is divided into small cells, and the sensors in the same cell cooperate to estimate the number of "1"s in the cell. Since the energy consumption is our major concern, we also use data-aggregation to further reduce the energy consumption. Note that only the number of "1"s instead of the individual measurements of each sensor are demanded at the fusion center. Our algorithm forms a tree rooted at the fusion center, and the counting results are aggregated and transmitted along the rooted-tree in a multi-hop fashion. Assuming that each sensor uses r^α units of energy to transmit each bit, where r is the transmission range of the sensor. We first show that the transmission energy consumption is at least $\Omega\left(\left(\sqrt{\frac{\log n}{n}}\right)^\alpha\right)$, and then propose a distributed function computation algorithm whose energy consumption is $O\left(n(\log\log n)\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right)$. The contents of Section 1.2 are a more detailed version of the results presented in [14].

In Section 1.3, we consider a different scenario where wireless channels are assumed to be reliable, but sensors are densely deployed so that the sensor measurements are highly correlated. Due to the broadcast nature of wireless channels, a transmission from one sensor can be heard by other sensors in its neighborhood. We consider a collocated network [5] where a transmission can be heard by all sensors in the network, and investigate the average energy consumption of real-time communication where the sensors send the measurements at each time slot without accumulating block measurements. We show that the exploitation of correlation can lead to a further reduction of energy consumption, and propose a methodology to analyze and design minimum cost communication and computation in wireless sensor networks. The results in Section 1.3 are somewhat preliminary since the proposed algorithm could be computationally very expensive to implement. However, our hope is that, by making a connection to stochastic control and dynamic programming (DP), one may be able to use results from the vast literature on approximate solutions to DP problems to ease the computational burden.

1.2 Distributed Function Computation in Noisy Wireless Sensor Networks

1.2.1 Notation

The following notation is used throughout this chapter. Given a sequence of random variables $X(n)$ indexed by n, and positive function $f(n)$, we will say that

(i) $X(n) = O(f(n))$, when there exists a positive constant c such that

$$\lim_{n\to\infty} \Pr(X(n) \le c f(n)) = 1 \text{ holds.}$$

(ii) $X(n) = \Omega(f(n))$ when there exists a positive constant c such that

$$\lim_{n\to\infty} \Pr(X(n) \ge c f(n)) = 1 \text{ holds.}$$

(iii) $X(n) = \Theta(f(n))$ when both $X(n) = \Omega(f(n))$ and $X(n) = O(f(n))$ hold.

Note that the above definitions also apply in the obvious way to deterministic functions.

1.2.2 Model

We consider a random network of n sensors that are uniformly and independently distributed on a unit square. Upon the occurrence of a certain event, sensor k records b_k, where b_k can take a value either “1” or “0.” The sensors have the capability to transmit this data over noisy wireless channels, and based on the data transmitted by the sensors in the network, a fusion center tries to evaluate some symmetric function $f(b_1,\ldots,b_n)$, i.e., a function which has the property that

$$f(b_1,\ldots,b_n) = f(\sigma(b_1,\ldots,b_n)),$$

for any permutation σ. Symmetric functions form a large class of functions, which includes almost all statistical functions like max, min, mean, etc. A key property of a symmetric function is that the function value only depends on the frequency-histogram; for the binary measurement case, the function only depends on the total number of "1"s in the network. So in this section, we will design algorithms to count the number of "1"s in the sensors' measurements, i.e., to compute

$$\sum_{i=1}^{n} b_i.$$

Since counting and computation are equivalent for symmetric functions, we will interchangeably use the terms counting and computation in this section.

Let S_i denote the location of sensor i and $|S_i - S_j|$ denote the distance from sensor i to sensor j. We use the protocol model in [7] for wireless interference with some additional assumptions.

(1) All nodes use the same transmission radius r, and the transmission power of one bit is r^α.
(2) A transmission from sensor i can be received at sensor j only if $|S_i - S_j| \leq r$ and $|S_k - S_j| \geq (1+\Delta)r$ for each sensor $k \neq i$ which transmits at the same time, where Δ is a protocol-specified guard-zone to prevent interference.
(3) A binary modulation scheme is used so that each transmission is either 1 or 0.
(4) Even if a transmission is received at the receiver, there is some probability $p < 1/2$ with which the received bit is flipped, i.e., the channel is a binary symmetric channel with error probability p.

Note that this model only holds when the near-field effects are negligible, which is assumed in this section.

By a counting algorithm, we mean a set of protocols (which may depend on n) to convey the appropriate information from the sensors to the fusion center and a protocol at the fusion center to use the received information to compute the number of "1"s in the sensor measurements. Given an algorithm for counting, we define the energy required by the algorithm to be the maximum energy required for the computation over all possible values of the measurements. Our goal is to characterize the minimum energy required subject to the constraint that the probability of error in the computation in a random network with n sensors goes to zero as $n \to \infty$. We only consider the transmission energy used for counting, and assume other energy expenditure, like energy used for computation, receiving, coordination etc, is negligible.

1.2.3 A Trivial Lower Bound on the Energy Consumption

The network needs to be connected to guarantee accurate function computation. Let $\mathscr{C}(n,r)$ denote the event that a random network with n nodes and common transmission radius r is connected, we first have following lemma.

Lemma 1. *Suppose that* $r \leq \frac{1}{6}\sqrt{\frac{\log n}{2n}}$. *Then given* $\varepsilon > 0$, *there exists* n_0 *such for any* $n \geq n_0$, *we have*

$$\Pr(\mathscr{C}(n,r)) \leq \varepsilon.$$

Proof. We divide the unit square into small cells with side length $\sqrt{\log n/(2n)}$, and each cell is further divided into nine min-cells as in Figure 1.1. Since $r \leq \frac{1}{6}\sqrt{\frac{\log n}{2n}}$, it

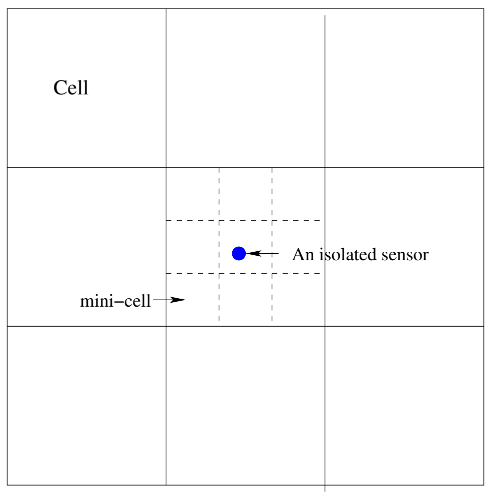

Fig. 1.1. An Example of An Isolated Sensor

is easy to see that the sensors in the central mini-cell of a cell can only communicate with the sensors in the same cell. If a cell contains only one sensor and the sensor is positioned in the central mini-cell, then the sensor is said to be isolated and the cell is said to be an isolated cell. Let N_p denote a Poisson random variable with mean n. Suppose that there are N_p sensors in the unit square, then the number of sensors in a cell is a Poisson random variable with mean $\log n/2$ and the probability that a cell is isolated is

$$\frac{1}{9} \times \frac{\log n}{2\sqrt{n}},$$

where $\log n/(2\sqrt{n})$ is the probability that the number of sensors in the cell is one, and $1/9$ is the probability that the sensor happens to be in the central mini-cell. Note that the numbers of sensors in each cell are i.i.d. Poisson random variables if the number of sensors in the unit square is N_p, and the probability that the network is connected increases with the number of sensors in the unit square. Thus, we have

$$\begin{aligned}\Pr(\mathscr{C}(n,r)) &\leq \Pr(\mathscr{C}(N_p,r)|N_p \geq n)\\ &\leq \frac{\Pr(\mathscr{C}(N_p,r))}{\Pr(N_p \geq n)}\\ &\leq 2\Pr(\mathscr{C}(N_p,r))\\ &\leq 2\left(1-\frac{1}{18}\frac{\log n}{\sqrt{n}}\right)^{\frac{n}{2\log n}},\end{aligned}$$

where the third inequality holds since $\Pr(N_p \geq n) \geq 1/2$. It is easy to see that

$$\left(1-\frac{1}{18}\frac{\log n}{\sqrt{n}}\right)^{\frac{n}{2\log n}}$$

goes to zero when n goes to infinity, so lemma holds.

We would like to comment that in [6], it has been shown that the network is asymptotically disconnected if $r \leq \sqrt{(\log n + c(n))/(\pi n)}$ and $\limsup_n c(n) < \infty$. This lower bound on r is tighter than the one in Lemma 1. However, Lemma 1 captures the lower bound up to the right order, and we have presented the proof is much easier and the result is sufficient for the purpose of the next lemma.

Next it is obvious that, in the worst-case when all sensors have a "1," each sensor has to broadcast its value once. Thus, we have the following trivial lower bound on the energy consumption.

Lemma 2 (A Trivial Lower Bound). *The minimum total transmission energy required to count is*

$$\Omega\left(n\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right) \tag{1.1}$$

.

Proof. Connectivity of the network is a necessary condition of correct counting. To guarantee connectivity, it has been shown in Lemma 1 that the transmission range of the sensors should be chosen as $\Omega\left(\sqrt{\frac{\log n}{n}}\right)$. Thus, the energy used per sensor transmission is $\Omega\left(\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right)$. There are n sensors in the network, each of which must make at least one transmission; thus, the total transmission energy required is $\Omega\left(n\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right)$.

1.2.4 An Upper Bound on the Energy Consumption

In this subsection, we propose a counting algorithm whose energy consumption is only a factor of $\log\log n$ more than the lower bound. We first present two well-known results for the reader's convenient reference. First, we study the error probability when using repetition coding. Consider a binary symmetry channel with error probability p where each bit is transmitted m times, and the receiver decodes the data using majority rule. Then we have following well-known bound [3] on the error probability, where the proof is provided for completeness.

Lemma 3. *Suppose one bit of data is transmitted m times over a binary symmetric channel with error probability p, and the receiver decodes the bit using majority rule. Then, the probability of decoding error is no greater than*

$$(4p(1-p))^{\frac{1}{2}m}.$$

Proof. Define m independent binary random variables $\{I_i\}$, where $I_i = 0$ with probability p and $I_i = 1$ with probability $1-p$. Using Chernoff's bound, we have

$$\Pr\left(\sum_{i=1}^{m} I_i < \frac{m}{2}\right) \le e^{\frac{m}{2}\log(4(1-p)p)} = (4p(1-p))^{\frac{1}{2}m}.$$

We also need the following coding theorem [3] for discrete memoryless channels for our analysis.

Theorem 1 (Gallager's Coding Theorem) *For any discrete memoryless channel with capacity C, any positive integer N, and any positive $R < C$, there exist block codes with $M = 2^{NR}$ codewords of length N for which the decoding error probability of each codeword is less than $4e^{-NE_r(R)}$, where $E_r(R)$ is a non-increasing function of R.*

□

Now, we consider the counting problem in detail. We first define the routing strategy. To transmit sensor information to the fusion center, we divide the unit square area into a regular lattice of B cells, and fix the transmission range

$$r = \sqrt{\frac{8}{B}}, \tag{1.2}$$

which guarantees a sensor can reach any other sensors within adjacent (common edge or corner) cells. We then adopt the hierarchical architecture of [5].

Routing Strategy: For each cell, we choose one sensor as the cell-center. Designating the fusion center as the root, we form a rooted tree like Figure 1.2, whose vertices include all the cell-centers, and whose links can only be between cell-centers of adjacent (common edge or corner) cells. Sensors first transmit data to the cell-centers, and then the data are aggregated and transmitted along the rooted-tree to the fusion center.

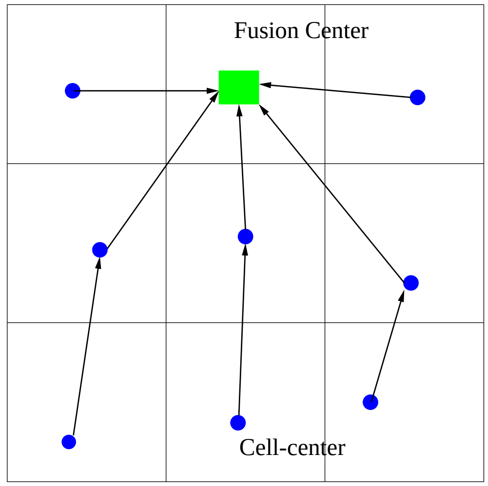

Fig. 1.2. A Wireless Sensor Network

We let $P(i)$ denote the parent of cell-center i, $C(i)$ denote the set of the children of cell-center i in the rooted tree, $H_{\max}$ denote the depth of the tree, and $H(i)$ denote the depth of the cell-center i in the tree ($H(\text{fusion center}) = 0$).

Note that the network is connected and the routing strategy is feasible if there is at least one sensor in each cell. It is easy to see that

$$E[\text{Number of sensors in each cell}] = \frac{n}{B}.$$

In [9, 13, 12], it has been shown that the number of sensors in each cell is n/B with high probability when $B = O\left(\frac{n}{\log n}\right)$. Choosing

$$B = \left(\left\lfloor \frac{n}{c_1 \log n} \right\rfloor\right)^2 \tag{1.3}$$

we bound the number of nodes in each cell in the following lemma, where the bounds are slightly tighter than the ones obtained in [12].

Lemma 4. *Suppose that the unit square is partitioned into B square cells, where B is chosen as in (1.3), and further let n_i denote the number of sensors in cell i. Then, for large enough n,*

$$Pr\left(\frac{c_1 \log n}{2} \leq n_i \leq 2c_1 \log n \quad \forall i\right) \geq 1 - \frac{2n^{\left(1-\frac{c_1}{8}\right)}}{c_1 \log n}. \tag{1.4}$$

Proof. Consider cell i. Note that the probability that a particular sensor is positioned in the cell is given by $(c_1 \log n)/n$. From the Chernoff's bound [11], we have that

$$\Pr(n_i \geq 2c_1 \log n) \leq e^{-c_1 \log n/3}$$

and

$$\Pr\left(n_i \leq \frac{c_1}{2}\log n\right) \leq e^{-c_1 \log n/8},$$

which implies that

$$\Pr\left(\frac{c_1}{2}\log n \leq n_i \leq 2c_1 \log n\right) \geq 1 - 2e^{-c_1 \log n/8}.$$

So from the union bound, we have

$$\Pr\left(\frac{c_1}{2}\log n \leq n_i \leq 2c_1 \log n \quad \forall i\right) \geq 1 - 2\frac{n}{c_1 \log n} e^{-c_1 \log n/8} = 1 - \frac{2n^{\left(1-\frac{c_1}{8}\right)}}{c_1 \log n}.$$

From above lemma, we know that if $c_1 \geq 8$, $\max_i n_i = O(\log n)$ and $\min_i n_i = \Omega(\log n)$ both hold. Throughout this section, B is chosen as in (1.3) with $c_1 = 8$, giving

$$r = 8\sqrt{\frac{\log n}{n}}.$$

Thus, the probability that routing strategy is feasible approaches 1 as n goes to infinity. In the following sections, we propose a distributed algorithm which works when each cell has at least one sensor. We assume that the algorithms report an error if the assumption does not hold.

Note that the wireless transmissions in neighboring cells will interfere with each other, so we adopt the cell scheduling scheme used in [7, 5]. Without loss of generality, we assume that $\Delta = 0.05$.

Cell Scheduling: We group every 5×5 cells into a super-cell, and index the cells from 1 to 25 as in Figure 1.3. We divide each time slot into 25 mini-slots, and at mini-slot i, the mini-cells with index i are chosen to be active, for example, all mini-cells with index 1 (as in Figure 1.3) are active in the first mini-slot of every time slot. When a cell is active, one sensor in the cell could be selected to transmit. In our algorithms, transmissions will occur only within a cell or between neighboring cells. Thus, it is easy to verify that there is only one transmitter within a distance $1.05r$ for each receiver, and simultaneous transmissions do not interfere with each other under the cell scheduling.

Now given that the routing strategy and cell scheduling, we will define protocols for intra-cell and inter-cell information processing and data aggregation. The protocols will have two distinct parts:

(1) Intra-Cell-Protocol: The information within cells is aggregated at the respective cell-centers.
(2) Inter-Cell-Protocol: The information aggregated by cell-centers is transmitted, and aggregated further, along the rooted tree to the fusion center.

We use the idea in [4] to design an algorithm for which the energy consumed is $\Theta\left(n(\log\log n)\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right)$. In wireless sensor networks, transmissions by a sensor can be heard by any sensor within its transmission range. Suppose there are $\tilde{n}$

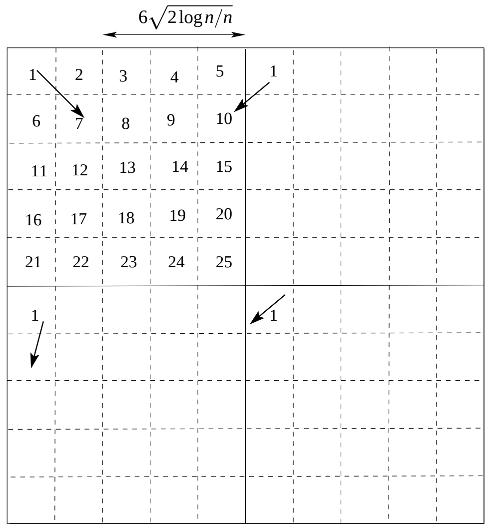

Fig. 1.3. Cell Scheduling ($\Delta = 0.05$))

sensors in sensor k's transmission range, then there are $\tilde{n}$ independent receptions for each bit sent by sensor k. The main idea in [4] is to use the reception diversity to obtain a good estimate of the bit transmitted by sensor k. But it requires additional transmissions among sensors; for example, it takes $\tilde{n}$ more transmissions for $\tilde{n}$ sensors to report the bits they received from sensor k. We will show how to use in-network processing to reduce the number of transmissions required to exploit the reception diversity.

Recall that b_k is the bit sensor k has. For cell i, define Δ_i as the set of indices of the sensors in cell i, and γ_i as the counting of cell-center i, so

$$\gamma_i = \sum_{k\in\Delta_i} b_k$$

if the counting is correct. For easy reference, we also define $\lambda = -\log(4p(1-p))$.

Counting-Algorithm-I:

When the a cell is active, the sensors first transmit and process measurements according to Intra-cell-protocol-I, and then the counting results are aggregated and transmitted according to Inter-cell-protocol-I.

Intra-Cell-Protocol-I (At cell i):

(i) The sensors in cell i take turns to transmit their bits. When it is the turn of sensor k, it broadcasts its bit $\left\lceil \frac{4}{\lambda}(\log\log n) \right\rceil$ times. Then, all other sensors in the cell will receive $\left\lceil \frac{4}{\lambda}(\log\log n) \right\rceil$ bits from sensor k. Sensor j ($j \neq k$) sets α_{jk} to be the majority of the bits received from sensor k, and sets A_j to be

$$A_j = b_j + \sum_{k\in\Delta_i, k\neq j} \alpha_{jk}$$

after all sensors broadcast their bits.

(ii) Select $\left\lceil \frac{n_i}{\log\log n} \right\rceil$ sensors in the cell. Each selected sensor j represents A_j using $\lceil \log_2 n_i \rceil$ bits, codes it using a block code with rate R_1 such that $E_r(R_1)/R_1 \geq 1$, and then broadcasts A_j once.

(iii) Suppose $\tilde{A}_j$ is the output of the binary symmetric channel between the cell-center and sensor j with input A_j. Cell-center i sets γ_i to be any mode of sequence $\{\tilde{A}_j\}$.

Inter-Cell-Protocol-I:

Define η_i to be the aggregated information of the subtree rooted at cell-center i. When cell-center i is scheduled, cell-center i sets

$$\eta_i = \gamma_i + \sum_{j \in C(i)} \tilde{\eta}_j,$$

where $\tilde{\eta}_j$ is the output of the channel between cell center j and cell center i with input η_j. Since $0 \leq \eta_i \leq n$, note that η_i can be represented using $\lceil \log_2 n \rceil$ bits. If i is the fusion center, then $\gamma_c = \eta_i$. Otherwise, it transmits η_i to cell-center $P(i)$ using a block code with rate R_2 such that $E_r(R_2)/R_2 > 1$.

We now analyze the energy requirement of Counting-Algorithm-I. First, in Lemma 5, we show that under Intra-Cell-Protocol-I,

$$\Pr\left(\text{All } \gamma_i \text{ are correct } \left| \frac{c_1}{2} \leq \frac{n_i}{\log n} \leq 2c_1 \ \forall i \right.\right) \geq 1 - \frac{1}{c_1 \log n}.$$

Then, in Lemma 6 and Theorem 2, we show that

$$\Pr\left(\gamma_c \text{ is correct } | \gamma_i \text{ is correct } \forall i\right) \geq 1 - \frac{4}{c_1 \log n}.$$

Finally, Theorem 2 quantifies the energy requirement of Counting-Algorithm-I.

Lemma 5. *Suppose* $\frac{c_1}{2}\log n \leq n_i \leq 2c_1 \log n$ *for all* i*. Then, by executing Intra-Cell-Protocol-I, the cell-centers can obtain* γ_i *with*

$$Pr\left(\gamma_i = \sum_{k \in \Delta_i} b_k \ \forall i \left| 2c_1 \geq \frac{n_i}{\log n} \geq \frac{c_1}{2} \ \forall i \right.\right) \geq 1 - \frac{1}{c_1 \log n} \tag{1.5}$$

and the number of transmissions required in each cell is $\Theta\left((\log n)(\log\log n)\right)$.

Proof. In the following analysis, we assume $\frac{c_1}{2}\log n \leq n_i \leq 2c_1 \log n$ holds for all i. First, the number of transmissions in each cell under Intra-Cell-Protocol-I is

$$n_i \left\lceil \frac{4}{\lambda}(\log\log n) \right\rceil + \left\lceil \frac{n_i}{\log\log n} \right\rceil \lceil \log_2 n_i \rceil = \Theta((\log n)(\log\log n)).$$

Next we investigate the probability that γ_i is correct, i.e., $\gamma_i = \sum_{k \in \Delta_i} b_k$. From Lemma 3, we have

$$\Pr\left(\alpha_{jk}=b_k\right)\geq 1-(4p(1-p))^{\frac{2\log\log n}{\lambda}}.$$

Note that A_j is correct if α_{jk} is correct for all $k\in\Delta_i$. From the union bound, we have

$$\Pr\left(A_j=\sum_{k\in\Delta_i}b_k\right)\geq 1-n_i(4p(1-p))^{\frac{2\log\log n}{\lambda}}\geq 1-\frac{2c_1}{\log n}.$$

Consider step (ii) of Intra-Cell-Protocol-I, from Theorem 1,

$$\Pr\left(\tilde{A}_j=A_j\right)\geq 1-4e^{-\frac{Er(R_1)}{R_1}\log_2 n_i}\geq 1-4e^{-\log\log n},$$

where the last inequality holds because $n_i\geq\frac{c_1}{2}\log n$. Thus,

$$\Pr\left(\tilde{A}_j=\sum_{k\in\Delta_i}b_k\right)\geq 1-\frac{2c_1+4}{\log n}.$$

Note that $\{\alpha_{jk}\}$ are i.i.d. for all $j\in\Delta_i$, so $\{A_j\}$ are identical and $\{\tilde{A}_j\}$ are i.i.d.. Now define i.i.d. random variables $\{I_j\}$ such that $I_j=1$ if $\tilde{A}_j=\sum_{k\in\Delta_i}b_k$, and $I_j=0$ if $\tilde{A}_j\neq\sum_{k\in\Delta_i}b_k$. Since γ_i is the mode of $\{\tilde{A}_j\}$, from Lemma 3, we have

$$\begin{aligned}\Pr\left(\gamma_i\neq\sum_{k\in\Delta_i}b_k\right)&\leq\Pr\left(\sum_j I_j<\frac{1}{2}n_i\right)\\&\leq\left(4\left(\frac{4c_1+4}{\log n}\right)\left(1-\frac{4c_1+4}{\log n}\right)\right)^{\frac{n_i}{2\log\log n}}\\&\leq e^{-(\log\log n-\log(16c_1+16))\frac{n_i}{2\log\log n}}\\&\leq e^{-\log n}.\end{aligned}$$

There are at most $\frac{n}{c_1\log n}$ cells in the network, so

$$\Pr\left(\gamma_i=\sum_{k\in\Delta_i}b_k\ \forall i\right)\geq 1-\frac{n}{c_1\log n}e^{-\log n}=1-\frac{1}{c_1\log n},$$

and the lemma holds.

Now, suppose that all γ_i are correct. Since η_i can be represented using $\lceil\log_2 n\rceil$ bits, each cell-center has $\lceil\log_2 n\rceil$ bits to transmit under Inter-Cell-Protocol-I.

Lemma 6. *Suppose all cell-centers have the correct γ_i, then under Inter-Cell-Protocol-I, the probability that the fusion center obtains the correct γ_c is bounded as follows:*

$$Pr\left(\gamma_c=\sum_k b_k\,\middle|\,\gamma_i=\sum_{k\in\Delta_i}b_k\ \forall i\right)\geq 1-\frac{4}{c_1\log n},\tag{1.6}$$

and the number of transmissions required is $\Theta(n)$.

Proof. Suppose all cell-centers have the correct γ_i, then $\gamma_c = \sum_k b_k$ if all η_i's are correctly received. From Theorem 1, there exists a block code satisfying the conditions given in step (i) of Inter-Cell-Protocol-I. Thus, for a given i,

$$\begin{aligned}\Pr(\eta_i \text{ is correctly received}) &\geq 1 - 4e^{-\frac{E_r(R_2)}{R_2}\log_2 n} \\ &\geq 1 - 4e^{-\log n},\end{aligned}$$

and from the union bound,

$$\begin{aligned}&\Pr\left(\gamma_c = \sum_k b_k \middle| \gamma_i = \sum_{k\in\Delta_i} b_k \ \forall i\right) \\ &= \Pr\left(\text{All } \eta_i\text{'s are correctly received} \middle| \gamma_i = \sum_{k\in\Delta_i} b_k \ \forall i\right) \\ &\geq 1 - \frac{4n}{c_1 \log n} e^{-\log n} \\ &= 1 - \frac{4}{c_1 \log n}.\end{aligned}$$

From Lemma 5 and Lemma 6, we have shown that, under Counting-Algorithm-I, the number of sensors in state “1” can be counted accurately with high probability when the number of sensors is large enough. Based on that, we have following theorem, which provides an upper bound on the energy requirement to solve our counting problem.

Theorem 2 *The number of sensors in state “1” can be counted accurately with high probability by total transmission energy consumption*

$$O\left(n(\log\log n)\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right),$$

and Counting-Algorithm-I is an asymptotically correct algorithm that achieves this energy consumption. Specifically, the probability of computation error at the fusion center is upper bounded by $\frac{7}{c_1 \log n}$.

Proof. Recall that

$$c_1 > \max\left\{8, \frac{4}{\lambda}\right\}.$$

From inequalities (1.4), (1.5) and (1.6), we have

$$\Pr\left(\gamma_c = \sum_k b_k\right) \geq 1 - \frac{7}{c_1 \log n},$$

which converges to one when n goes to infinity. So Counting-Algorithm-I is asymptotically correct.

Further, from Lemma 5 and Lemma 6, the number of transmissions under Counting-Algorithm-I is $\Theta(n(\log\log n))$. Since the common transmission range is $\sqrt{\frac{8c_1 \log n}{n}}$, the total energy consumption is

$$\Theta\left(n(\log\log n)\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right). \tag{1.7}$$

The theorem holds because there may exist other algorithms that consume less energy.

A simple lower bound has been obtained in Lemma 2. Comparing it with the upper bound in Theorem 2, we can see that the upper bound differs by a factor of *only* $(\log\log n)$ from the lower bound. But it is still not clear how good our bound is. A more general computational problem than ours, i.e., one of knowing all the bits in the network, is considered for a broadcast network in [4]. The number of transmissions required there is also shown to be $O(n(\log\log n))$. This suggests that one may be able to improve our upper bound on the energy usage since counting is easier than detecting all the bits in the network. On the other hand, parity computation which is a simpler problem than counting is also studied in [4], but the number of transmissions needed is again $O(n(\log\log n))$, the same complexity as Counting-Algorithm-I. To the best of our knowledge, this is the best upper bound in the literature for parity computation in broadcast networks. Further, our network with its multihop architecture also requires more transmissions for the data from the sensors to reach the fusion center. This suggests that our upper bound on energy usage is quite good.

In this section, we presented the simplest case, where each sensor has only one binary measurement to report, to demonstrate that energy savings can be achieved by sensor collaboration and data aggregation. More general cases can be found in [14] and [15]. In [14], besides the simplest case, we also studied the case where each sensor has N binary measurements to report and the symmetric function needs to be computed for each measurement. We showed that the total transmission energy consumption can be reduced to $O\left(n\left(\max\left\{1, \frac{\log\log n}{N}\right\}\right)\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right)$ per measurement. When $N = \Omega(\log\log n)$, the energy consumption is $\Theta\left(n\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right)$ per measurement, which is a tight bound. We also considered the case that we only want to know roughly, i.e., how many sensors have "1." The answer can be obtained with the transmission energy consumption $\Theta\left(n\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right)$. All these results can be extended the cases where the sensor measurements taken value from $\{0,\ldots,m-1\}$, and the details are presented in [15].

1.3 Minimum Cost Real-time Function Computation in Wireless Sensor Networks

In the previous section, we investigated function computation in wireless sensor networks, where we did not make any assumption on data correlation. However, in the case where sensors are densely deployed, the sensor measurements could be highly correlated. This additional energy savings can be achieved by exploiting such a data correlation. Further, in the previous problem, we assumed that sensors use the same amount of energy to transmit a “0” or “1.” In this section, we will consider the case where the sensor data is correlated, communication costs can be different, and transmission channels need not not be binary. For example, silence could be used to convey information and it is a zero-cost signal. Then to reduce the energy consumption, we should use low-cost signals as frequently as possible. In the following simple example, we illustrate that energy savings can be achieved by exploiting the data correlation.

1.3.1 A Simple Example

Consider a network with three sensors each with a binary measurement value. The observations are random variables and their joint distribution is given in in Table 1.1.

Event	Probability
0 0 1	$\frac{1}{6}$
0 1 0	$\frac{1}{6}$
0 1 1	$\frac{1}{6}$
1 0 1	$\frac{1}{3}$
1 1 0	$\frac{1}{6}$

Table 1.1. Joint Distribution of Sensor Observations

An event in the table is a particular set of possible observations. For example, the first row indicates that the probability of sensor 1 observing a “0,” sensor 2 observing a “0”, and sensor 3 observing a “1” is $1/6$. Note that the unrealistic events have zero probability to occur.

Assume that the sensor can use two transmission signals $\{S,P\}$ to convey the information, where S indicates silence, with cost zero; and P is some pulse single with energy cost E. Now we consider following three different cases.

(1) Now suppose the sensors are deployed far away from each other, so no sensor can hear the others’ transmissions. For each sensor, it either use S to represent 0 or 1. Let F_i denote the encoding scheme. It is easy to see the minimum cost scheme is as in Table 1.2, and the expected cost is $4E/3$.

Sensor	Encoding Scheme
Sensor 1	$F_1(0) = S$ and $F_1(1) = P$
Sensor 2	$F_2(0) = S$ and $F_2(1) = P$
Sensor 3	$F_3(0) = P$ and $F_3(1) = S$

Table 1.2. Case 1 Encoding Scheme: Isolated Sensors

(2) The sensors do not know others' observations, but can hear each other. After one sensor transmits its information, all other sensors can update their probability of the various possible events, and use this information to reduce their transmission costs. The sensors cannot simultaneously transmit, but transmit in sequential order. In this collocated case, the encoding schemes of sensor 2 and 3 depend on what they heard sensor 1, and sensor 1 and 2, transmit respectively. Thus, use the transmission scheme is in Table 1.3. It is easy to see that each event associates a distinct transmission signals and the expected cost is E.

Sensor	Encoding Scheme
Sensor 1	$F_1(0) = S$ and $F_1(1) = P$
Sensor 2	$F_2(S,0) = P$, $F_2(S,1) = S$, $F_2(P,0) = S$, and $F_2(P,1) = P$
Sensor 3	$F_3(S,P,1) = S$, $F_3(S,S,0) = P$, $F_3(S,S,1) = S$ $F_3(P,S,1) = S$, and $F_3(P,P,0) = S$

Table 1.3. Case 2 Encoding Scheme: Collocated Network

We can see that, even for this simple example, significant energy savings (25%) can be achieved. Next, we propose a stochastic control approach to obtain the minimum cost transmission scheme.

1.3.2 Model

Consider a wireless sensor network consisting of n sensors, and each sensor has a measurement X_i where X_i taking values from $\mathscr{X}$, and $\mathscr{X}$ is the sample space of the observations, and $|\mathscr{X}| = m$. The network has a special node called the fusion center whose goal is to compute some function $G(\mathbf{X})$ based on sensor observations $\mathbf{X}$. We consider a collocated network [5] where a transmission can be heard by all sensors in the network, and only one transmission is allowed at one time. Let x_i denote a realization of X_i, and $x_i(1),\ldots,x_i(k)$ denote different realizations of X_i. Also, let $\mathbf{X}_i^j$ denote $(X_i,\ldots,X_j)$, $\mathbf{x}_i^j$ be a realization of $\mathbf{X}_i^j$, and $\Pr\left(\mathbf{x}_i^j\right)$ be the abbreviation of $\Pr\left(\mathbf{X}_i^j = \mathbf{x}_i^j\right)$.

In this section, we assume that sensors sequentially take their turn to transmit, and the order is fixed such that sensor 1 transmits first, then sensor 2 and so on, as in Figure 1.4. Each sensor only transmits once, after which the fusion center needs to

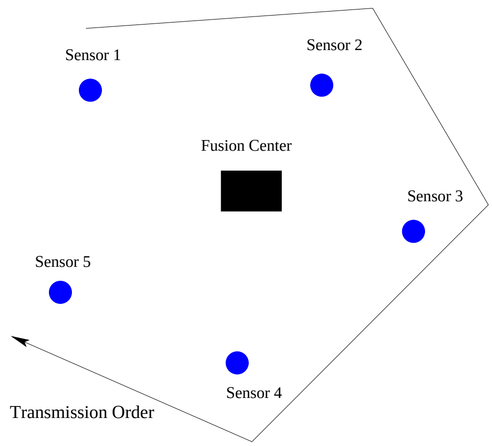

Fig. 1.4. The Order of Sensor Transmissions

compute $G(\mathbf{X})$. Note that the fusion center does not need to recover $\mathbf{X}$, it only needs to know enough information to compute the function value. Let C_i denote the signal node i transmits, and c_i be a realization of C_i. We assume $c_i \in \mathscr{S}$, where $\mathscr{S}$ is the set of signals and $|\mathscr{S}| = m$. For simplicity, we let c_i denote both the name and cost of the signal. Furthermore, it is easy to see that C_i is a function of the transmissions before node i and the measurement X_i of node i. Accordingly, we define an encoding function F_i such that

$$C_i = F_i(\mathbf{C}_1^{i-1}, X_i).$$

Further define $\mathbf{F}_1^i$ such that

$$\mathbf{F}_1^i(\mathbf{X}_1^i) = (F_1(X_1), F_2(F_1(X_1), X_2), \ldots, F_i(\mathbf{F}_1^{i-1}(\mathbf{X}_1^{i-1}), X_i)),$$

and let $\mathbf{F}$ denote $\mathbf{F}_1^n$. An encoding scheme $\mathbf{F}$ is said to be feasible if

$$\mathbf{F}(\mathbf{x}) \neq \mathbf{F}(\mathbf{y}),$$

for every $\mathbf{x}$ and $\mathbf{y}$ such that $\Pr(\mathbf{x}) > 0$, $\Pr(\mathbf{y}) > 0$ and $G(\mathbf{x}) \neq G(\mathbf{y})$. Let $\mathscr{F}$ denote the set of feasible encoding functions, and $\sum \mathbf{F}(\mathbf{X}) = \sum_{i=1}^n F_i(\mathbf{F}_1^{i-1}(\mathbf{X}_1^{i-1}), X_i)$. Our goal is find $\mathbf{F}^*$ such that

$$\begin{aligned} \mathbf{F}^* &= \arg\min_{\mathbf{F}} E\left[\sum \mathbf{F}(\mathbf{X})\right] \\ &\text{subject to: } \mathbf{F} \in \mathscr{F}. \end{aligned} \tag{1.8}$$

1.3.3 Stochastic Control Approach

In this subsection, we propose a stochastic control approach to solve the minimum cost problem (1.8). We use superscript -1 to indicate decoding functions, for example, $\mathbf{F}^{-1}$ is the decoding function of $\mathbf{F}$ such that $\mathbf{F}^{-1}(\mathbf{c}) \in (2^{\mathscr{X}})^n$ is the preimage of

c. Note that decoding functions and known to all sensors and the fusion center. The decoding process is as follows: When the fusion center receives $\mathbf{c}_1^i$, it can decode $\mathbf{c}_1^i$ to obtain $F_i^{-1}(\mathbf{c}_1^i)$, which is a subset of $\mathscr{X}$. After all sensors transmit, the fusion center decodes the received information and obtains

$$\mathscr{Y} = (F_1^{-1}(c_1), \dots, F_n^{-1}(\mathbf{c}_1^n)).$$

Then the fusion center takes $G(\mathscr{Y})$ as the function computation result, which is unique and correct if the encoding function $\mathbf{F}$ is feasible. Similarly, sensor i can compute $(\mathbf{F}_1^{i-1})^{-1}(\mathbf{c}_1^{i-1})$ after receiving $\mathbf{c}_1^{i-1}$. Note that $(\mathbf{F}_1^{i-1})^{-1}(\mathbf{c}_1^{i-1}) \in (2^{\mathscr{X}})^{i-1}$. Let $\mathfrak{A}_i$ be a random variable taken values from $2^{\mathscr{X}}$, and $\mathfrak{a}_i$ be a realization of $\mathfrak{A}_i$. We can first design an encoding scheme $\tilde{\mathbf{F}}$ based on $\mathfrak{A}_1^{i-1}$ and X_i, and then obtain $\mathbf{F}$ as follows:

(1) First design $c_i = \tilde{F}_i(\mathfrak{a}_1^{i-1}, x_i)$ for each $(\mathfrak{a}_1^{i-1}, x_i)$ such that $\Pr(\tilde{\mathbf{x}}_1^{i-1}) > 0$ for all $\tilde{\mathbf{x}}_1^{i-1} \in \mathfrak{A}_1^{i-1}$, and $\Pr(x_i|\mathfrak{a}_1^{i-1}) > 0$.
(2) Let

$$F_i(\mathbf{C}_1^{i-1}, X_i) = \tilde{F}_i((\tilde{\mathbf{F}}_1^{i-1})^{-1}(\mathbf{C}_1^{i-1}), X_i). \quad (1.9)$$

Note that we impose conditions on $\mathfrak{a}_1^{i-1}$ and x_i in step (1) since we only need to design coding scheme for events with positive probability. The relation between F_i and $\tilde{F}_i$ is illustrated in Figure 1.5.

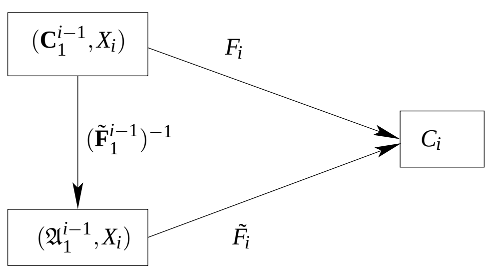

Fig. 1.5. The Relation between F_i and $\tilde{F}_i$

We next model the coding process as a controlled Markov chain $\{\mathfrak{Z}_i\}$ under decision process $\tilde{\mathbf{F}}$, where $\mathfrak{Z}_i = (\mathfrak{A}_1, \dots, \mathfrak{A}_{i-1}, \mathscr{X}, \dots, \mathscr{X})$. Note $\mathfrak{Z}_i$ represents the information obtained at the fusion center after first i sensors transmit. Next let $\mathfrak{z}_i$ be a realization of $\mathfrak{Z}_i$. The transition probability of the controlled Markov chain under decision $\tilde{\mathbf{F}}$ is defined to be

$$\begin{aligned}\Pr\left(\mathfrak{z}_{i+1} \middle| \mathfrak{z}_i, \tilde{F}_i\right) &= \Pr\left((\mathfrak{a}_1, \dots, \mathfrak{a}_i, \mathscr{X}, \dots, \mathscr{X}) \middle| (\mathfrak{a}_1, \dots, \mathfrak{a}_{i-1}, \mathscr{X}, \dots, \mathscr{X}), \tilde{F}_i\right) \\ &= \Pr\left(\mathfrak{a}_i \middle| \mathfrak{a}_1^{i-1}\right)\end{aligned}$$

if following conditions hold:

(1) $\Pr(\mathbf{x}_1^{i-1}) > 0$ for every $\mathbf{x}_1^{i-1} \in \mathfrak{a}_1^{i-1}$;
(2) $\mathfrak{a}_i = \tilde{F}_i^{-1}(\mathfrak{a}_1^{i-1}, c)$ for some $c \in \mathscr{S}$.

Otherwise $\Pr\left(\mathfrak{z}_{i+1} \,\middle|\, \mathfrak{z}_i, \tilde{F}_i\right) = 0$. Note that we impose the first condition since we only design coding scheme for events with positive probability, and impose the second condition to guarantee that $\mathfrak{a}_i$ will be conveyed by some signal. The cost of decision $\tilde{F}_i$ given $\mathfrak{z}_{i-1}$ is

$$C\left(\mathfrak{z}_i, \tilde{F}_i\right) = E\left[\tilde{F}_i\left(\mathfrak{a}_1^{i-1}, X_i\right)\right].$$

Since the function $G(\mathbf{X})$ needs to be computed accurately at the end of sensor transmissions, we impose penalty cost on $\mathfrak{Z}_n$ such that

$$C^p(\mathfrak{z}_n) = \begin{cases} \infty, & \text{if } \exists \mathbf{x}, \mathbf{y} \in \mathfrak{z}_n \text{ s.t. } G(\mathbf{x}) \neq G(\mathbf{y}); \\ 0, & \text{otherwise.} \end{cases}$$

Thus, from the definition of cost (1.10), we can see that problem (1.8) is equivalent to the following stochastic control problem

$$\min_{\tilde{\mathbf{F}}} E\left[C^p(\mathfrak{Z}_n) + \sum_{i=1}^{n} C(\mathfrak{Z}_i, \tilde{F}_i) \,\middle|\, \mathfrak{Z}_1 \right]. \tag{1.10}$$

So the minimum cost scheme $\tilde{\mathbf{F}}^*$ can be obtained by solving this standard stochastic control problem, and $\mathbf{F}^*$ can be further obtained from equation (1.9).

We would like to comment that the goal of this section is to model the minimum cost real-time function computation to a standard stochastic control problem, thus provide a methodology to design the minimum cost transmission scheme. Note that $\mathfrak{a}_i \in 2^{\mathscr{X}}$, so $\mathfrak{z}$ can take $2^{i|\mathscr{X}|}$ different values in general and the complexity of stochastic control problem (1.10) exponentially increases both in n and $|\mathscr{X}|$. So to make this approach applicable to large n or large $\mathscr{X}$, we need to further reduce the complexity, which is still an open problem for general cases. However, for a special case where $\Pr(\mathbf{x}) > 0$ for any $\mathbf{x} \in \mathscr{X}^n$ and $G(\mathbf{x}) \neq G(\mathbf{y})$ if $\mathbf{x} \neq \mathbf{y}$, a simple $\mathbf{F}^*$ can be solved from (1.10).

Theorem 3 *Suppose that* $\Pr(\mathbf{x}) > 0$ *for any* $\mathbf{x} \in \mathscr{X}^n$, $G(\mathbf{x}) \neq G(\mathbf{y})$ *if* $\mathbf{x} \neq \mathbf{y}$, *and* $c_i > c_j$ *if* $i > j$. *Given* $\mathbf{x}_1^{i-1}$, *we order* x_i *according to* $\Pr(x_i|\mathbf{x}_1^{i-1})$, *from the largest to the smallest, and let* $I(x_i|\mathbf{x}_1^{i-1})$ *denote the rank of* x_i. *Then the minimum cost scheme is*

$$F_i^*(x_i, \mathbf{c}_1^{i-1}) = c_{I(x_i|({F^*}_1^{i-1})^{-1}(\mathbf{c}_1^{i-1}))}. \tag{1.11}$$

Proof. Since $\Pr(\mathbf{x}) > 0$ for any $\mathbf{x} \in \mathscr{X}^n$ and and $G(\mathbf{x}) \neq G(\mathbf{y})$ if $\mathbf{x} \neq \mathbf{y}$, it is easy to see that $C^p(\mathfrak{z}_n) = 0$ if $\mathfrak{z}_n \in \mathscr{X}^n$, and $C^p(\mathfrak{z}_n) = \infty$ otherwise. Thus $\tilde{\mathbf{F}}^*$ needs to satisfy

$$\tilde{\mathbf{F}}^*(\mathbf{x}) \neq \tilde{\mathbf{F}}^*(\mathbf{y}) \text{ if } \mathbf{x} \neq \mathbf{y}, \tag{1.12}$$

which implies that

$$\tilde{F}_i^*(x_i, \mathbf{x}_1^{i-1}) \neq \tilde{F}_i^*(y_i, \mathbf{x}_1^{i-1}) \text{ if } x_i \neq y_i. \tag{1.13}$$

Now let $\tilde{\mathscr{F}}^*$ denote the set of $\tilde{\mathbf{F}}$ satisfying (1.12), and $\tilde{\mathscr{F}}_i^*$ denote the set of $\tilde{F}_i$ satisfying (1.13), then problem (1.10) can be re-written as

$$\begin{aligned}
&\min_{\tilde{\mathbf{F}}} E\left[C^p(\mathfrak{Z}_n) + \sum_{i=1}^n C(\mathfrak{Z}_i, \tilde{F}_i) \middle| \mathfrak{Z}_1\right] = \min_{\tilde{\mathbf{F}} \in \tilde{\mathscr{F}}^*} E\left[\sum_{i=1}^n C(X_i, \tilde{F}_i)\right] \\
&= \sum_{i=1}^n \sum_{\mathbf{x}_1^{i-1} \in \mathscr{X}^{i-1}} \left(\min_{\tilde{F}_i \in \tilde{\mathscr{F}}_i^*} \sum_{x_i \in \mathscr{X}} \tilde{F}_i(x_i, \mathbf{x}_1^{i-1}) \Pr\left(x_i | \mathbf{x}_1^{i-1}\right)\right) \Pr\left(\mathbf{x}_1^{i-1}\right).
\end{aligned}$$

Given $\mathbf{x}_1^{i-1}$, it is easy to see that the optimal $\tilde{F}_i$ is to assign the lower cost signal to the event with higher probability, so (1.11) is the optimal scheme.

1.4 Conclusion

In Section 1.2, we investigated counting problems in multi-hop networks with noisy communication channels. We considered the case where each sensor has a single measurement, and showed by construction that feasible algorithms exist whose energy consumption is class $O\left(n(\log\log n)\left(\sqrt{\frac{\log n}{n}}\right)^{\alpha}\right)$. In Section 1.3, we investigated the minimum cost real-time function computation in single-hop networks, and showed that the problem can be solved using a stochastic control approach.

References

1. A. Dhulipala, C. Fragouli and A. Orlitsky Silence based communication for sensor networks. In *IEEE International Symposium on Information Theory,* July 2006.
2. C. Fragouli and A. Orlitsky Silence is golden and time is money: power-aware communication for sensor networks, C. Fragouli and A. Orlitsky. In *the Allerton Conference,* September 2005.
3. R. G. Gallager. Information Theory and Reliable Communication. John Wiley & Sons, New York, 1968.
4. R. G. Gallager. Finding parity in a simple broadcast network. In *IEEE Transactions on Information Theory,* vol. 34, pp 176-180, 1988.
5. A. Giridhar and P. R. Kumar. Computing and communicating functions over sensor networks. In *IEEE Journal on Selected Areas in Communications*, pp. 755–764, vol. 23, no. 4, April 2005.
6. P. Gupta and P. Kumar. Critical power for asymptotic connectivity in wireless network. In *Stochastic Analysis, Contrtol, Optimization and Applicaitons: a Volume in Honor of W.H.Fleming,* W. McEneaney, G. Yin and Q. Zhang, Eds., 1998
7. P. Gupta and P. Kumar. The capacity of wireless networks. In *IEEE transactions of Information Theory,* vol. 46, no.2, pp. 388-404, 2000.
8. N. Khude, A. Kumar and A. Karnik. Time and Energy Complexity of Distributed Computation in Wireless Sensor Networks. In *Proceedings of the IEEE Infocom,* 2005.

9. S. R. Kulkarni and P. Viswanath. A Deterministic Approach to Throughput Scaling in Wireless Networks. In *IEEE Trans. on Information Theory,* Vol. 50, No.6, pp. 1041-1049, June 2004.
10. E. Kushilevitz and Y. Mansour. Computation in Noisy Radio Networks In *Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms,* pp. 236-243, 1998.
11. M. Mitzenmacher and E. Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge, 2005.
12. S. Toumpis and A. J. Goldsmith Large wireless network under fading, mobility, and delay constraints. In *Procedings of IEEE INFOCOM,* 2004.
13. F. Xue and P. R. Kumar. The number of neighbors needed for connectivity of wireless networks. Wireless Networks, pp. 169–181, vol.10, no. 2, March 2004.
14. L. Ying, R. Srikant and G. E. Dullerud. Distributed Symmetric Function Computation in Noisy Wireless Sensor Networks with Binary Data. In *Proc. 4rd Intl. Symposium on Modelling and Optimization in Mobile, Ad Hoc, and Wireless Networks,* April 2006.
15. L. Ying, R. Srikant and G. E. Dullerud. Distributed Symmetric Function Computation in Noisy Wireless Sensor Networks. Preprint. Available at http://www.ifp.uiuc.edu/~lying/Publicaitons/DSFComputation_J.pdf
16. Y. Zhu and R. Sivakumar Challenges: communication through silence in wireless sensor networks. In *Proceedings of the 11th annual international conference on Mobile computing and networking.*